

Finansal Veri Üzerine Bilgi Getirimi: KAP Bildirimleri ile Bir Çalışma

Muhammed Ali Karşlı
Bilgisayar Mühendisliği
Galatasaray Üniversitesi

Sena Özbelen
Bilgisayar Mühendisliği
Gebze Teknik Üniversitesi

Yağız Hakkı Aydın
Bilgisayar Mühendisliği
Gebze Teknik Üniversitesi

Özetçe—Teknolojinin geliştiği bir dönemde veri ve verinin analizi çok önemli bir yer kaplamaktadır. Özellikle de finansal sektörlerdeki verinin anlamlandırılması için birçok çalışma yapılmaktadır. Bu sektörün ana parçasını oluşturan şirketler ve onların durumları yapılan veri analizini de etkilemektedir. Bu kapsamda, Borsa İstanbul içerisinde yer alan şirketlerin bulunduğu Kamuyu Aydınlatma Platformu birçok bilgiye ev sahipliği yapmaktadır. Bu çalışmada, platform üzerinden yayınlanan bildirimlerin incelenerek kullanıcı tarafından sorulara cevap aranması üzerinde durulmuştur.

Anahtar Kelimeler—finansal veri analizi, bilgi getirimi, KAP bildirimleri.

I. GİRİŞ

Finansal veri analizi; finansal piyasaları anlamak, ekonomik hareketleri anlamlandırmak, şirketlerin performanslarını ölçmek, piyasa trendlerini tespit etmek ve yatırım fırsatlarını keşfetmek açısından varlığı inkar edilemez bir önem taşımaktadır.

Bu bağlamda, Kamuyu Aydınlatma Platformu (KAP), yatırımcılar için Türkiye’deki şirketler ve sermaye piyasalarıyla ilgili en değerli kaynaktır. İşbu kaynağın kullanımının kullanıcının isterlerine göre özelleştirilmesi ve iyileştirilmesi yatırımcıların karar verme mekanizmalarının verimliliğini arttıracak bir potansiyele sahiptir.

Finansal verilerin ayıklanması ve işlenmesi, birbirinden bağımsız problemler üzerine kompakt bir çözüm sunmayı gerektirmektedir. Bu prosedürler sonucunda elde edilen ürün, son kullanıcının - yatırımcının - rutin işlemlerini hızlandırabilecek ve kullanıcıya destek sağlayabilecek bir etkiye mükettir.

II. LİTERATÜRDEKİ ÇALIŞMALAR

Chatbot sistemleri üzerine yapılan literatür araştırmalarında ilk olarak hem kullanıcıdan gelen girdinin hem de kullanılacak belgelerin içerisinde yer alan metinlerin ön işlemlerinden bahsedilmektedir.

“Chatbot as a Virtual Assistant to Retrieve Information from Datasheets Using Memory Controllers Domain Knowledge” [1] adlı makalede sunucu tarafında pdf formatındaki belgelerin işlenebilir metin formatında çekilmesi için bazı pdf kütüphanelerinin kullanılması ön plandadır. Hem tabloların hem de metinlerin düzgün bir şekilde çekilmesi önem taşımaktadır.

Örneğin “UniRaG: Unification, Retrieval, and Generation for Multimodal Question Answering With Pre-Trained Language Models” [2] adlı makalede resim gibi farklı verilerle de

çalışma yapılmıştır. Bu tarz durumlarda da yine verinin metin haline çevrilmesi uygun görülmüştür.

Ardından da genel olarak chatbot sistemleri üzerine durulmuştur. Burada daha çok RASA framework yapısı tarzı teknolojilerle chatbot sistemi kurulmuştur. Bu sistemin içerisinde ise ilk olarak ilgili belgenin çekilmesi ve ardından da belge içerisinde ilgili cevabın bulunması yer almaktadır. Bazı makaleler benzerlik üzerine bir yaklaşım sergilemektedir. BM25 veya TF-IDF gibi metodlar kullanılmaktadır ve genel olarak kosinüs benzerliği ile benzerlik hesabı yapılmaktadır [1]. Benzerlik kullanılmadığı durumlarda büyük dil modellerinden faydalanılmaktadır. Genelde BERT gibi modeller tercih edilmektedir [2]. İlgili belgeden sorunun cevabını getirmek için de genel olarak soru ve ilgili kısım büyük dil modeline verilerek cevap elde edilir. Bu modellerin kullanılabilmesi için bir eğitim aşaması gerekmektedir.

III. METODOLOJİ

A. Veri Seti

KAP, Borsa İstanbul’da işlem gören şirketlerin, borsa ve sermaye piyasası mevzuatı kapsamındaki çeşitli araçların kamuya açıklanması gerekli bildirimleri duyurduğu elektronik sistemdir [3].

KAP’taki bildirim tipleri; finansal raporlar, özel durum açıklamaları, fon bildirimleri ve diğer bildirimler şeklinde dörde ayrılabilir. İşbu bildirimlere, KAP sorgu sisteminden ulaşılabilmektedir [4].

Her bir bildirim; başlık, tarih, bildirim türü, bildirimle ilgili şirketler, bildirim açıklaması gibi yinelenen yapılar ve bildirim gövdesi, tablolar gibi münferit yapılar sahiptir.

Bildirimler KAP’ın internet sitesi üzerinden paylaşılmakta olup, her bir bildirim pdf formatında, varsa eklerini de - genelde pdf veya xls formatında - indirmek mümkündür. Her bildirim, sistem tarafından, kendine özgü ve rasyonel bir anlam ifade etmeyen bir sayıyla adlandırılmıştır. Eğer varsa, bildirimin ekleri de, ilgili şirket tarafından verilen ve önceden tahmin edilmesi güç isimlerle adlandırılmıştır.

Veri seti işleminde ve proje genelinde python dili kullanılmıştır.

B. Veri Ön İşlemesi

KAP’ın bildirimlerine erişilmesini sağlayan halka açık bir API olmadığından ötürü bildirimlere otonom ulaşılması amacıyla requests kütüphanesi kullanılmıştır.

Verilerin saklanması için şirket bazlı bir depolama sistemi kurulmuş, bildirimleri indirilmek istenen şirket adında bir klasör açılmış, bu şirkete ait her bir bildirim ve bildirimin ekleri, bildirimin adında bir alt klasörde saklanmıştır.

Ulaşılan bildirimler, *beautifulsoap* kütüphanesi kullanılarak, internet sitesinin kodları üzerinden ayıklanmıştır. Bildirime ait başlık, açıklama, bildirim türü, ilgili şirketler gibi bilgiler; eğer varsa tablolar ve tablo benzeri yapılar bildirimin klasöründe saklanan bir *json* dosyasında depolanmıştır.

Veri indirilmesi için oluşturulan kod, toplu bildirim ve ek indirmenin yanı sıra, KAP bildirim sorgu sistemindeki gibi belli özelliklere (tarih aralığı, sektör, pazar, endeks vb.) sahip bildirimleri de bulup indirebilecek şekilde uyarlanmıştır. Bu sayede kullanıcının girdisiyle ilgili bildirimlerin tespiti sağlanabilmektedir.

pdf olarak yer alan ek belgelerde finansal tablolar sık sık görülmektedir ve bu tarz bilgiler düzgün bir şekilde çıkartılmalıdır. Yapılandırılmamış verilerin önceden tanımlanmış herhangi bir yapısı yoktur ve *pdf* formatı da bunun bir örneğidir bu yüzden veriyi düzgün bir şekilde ele almak kolay değildir. Hali hazırda *pdf* formatını ele alan birtakım kütüphaneler bulunmaktadır ama ek belgelerde çok iyi sonuçlar alınamamıştır bu yüzden "parsee-pdf-reader" [5] adlı bir GitHub projesinden faydalanıldı. Bu proje *pdf* içerisinde yer alan finansal/numerik tablolara odaklıdır. Bununla birlikte tablolar ve metinler belgeden çıkartılıp *json* dosyasında depolanmıştır.

C. Bilgi Getirimi Sistemi

1) *Kullanıcı Girdisi*: Kullanıcının istediği bildirimle alakalı soru bu kısımda ele alınır. İlk olarak bildirimden belli başlı anahtar kelimeler çekilmektedir. Şirket ismi, tarih ve bildirim tipi olabilecek her türlü kelime burada girdi üzerinden çekilir. Çekilen anahtar kelimeler ile kullanıcının bilgi istediği potansiyel bildirimler bulunmaktadır. Bu kısım benzerlik üzerine kurulmuştur.

Şirket isimlerini içeren bir *csv* dosyası yer almaktadır. Bu dosyada şirketlerin resmi isimlerinin yanı sıra, borsada yer alan isimleri ve kullanıcıların girebilecekleri muhtemel isim seçenekleri yer almaktadır. Örneğin *ASELSAN* için *ASELSAN ELEKTRONİK SANAYİ VE TİCARET A.Ş.*, *ASELS* ve *ASELSAN* isimleri bulunmaktadır. Bu noktada girdinin tamamı üzerinde bu isim aranmak zorundadır ama bu benzerlik sonucu çok fazla etkileyebilmektedir. Bu sorundan ötürü kullanıcı girdisi birli, ikili ve üçlü segmentlere ayrılmaktadır ve benzerlik bu segmentlerle gerçekleştirilmektedir. Bu kısımda fuzzy matching denilen bir teknik kullanılmaktadır. Fuzzy matching yapısında exact matching yapısı aksine tam bir eşleşme aranmaz. Edit distance üzerinden benzerlik ölçümü yaparak eşleşmeler yapar. Bu eşleşme sonucunda ilgili şirket bulunur.

Tarihler için genel olarak regular expression denilen yapıdan faydalanılır. Belli formattaki tarihler (01.01.2024 gibi) bu yapıyla çekilir. Aynı şekilde ay ismi sayıyla değil ismiyle yazıldığında da bu yapı kolaylıkla algılar. Bazı tam tarih içermeyen durumlar da bir liste halinde belirtilmiştir. Bu tarz durumlara örnek olarak "bu yıl", "geçen yılın ilk çeyreği" söylenebilir. Ayrıca "Mayıs 2024" veya "Mayıs ayı" gibi tarihleri de regular expression yardımıyla desteklemektedir. Tarih konusunda kurallara bağlı olması nedeniyle şirket ismi kadar bir esneklik sağlanamamıştır.

Bildirim tipinin bulunabilmesiyle alakalı kısımda da ilk olarak her bir bildirim tipi için anahtar kelimeler tanımlanmıştır. Bu kısımda da yine benzerlik üzerinden yapılmaktadır. Şirket ismi için *thefuzz* kütüphanesinden *extractOne* fonksiyonu kullanılırken bildirim tipi için *diffib* kütüphanesinden *get_close_matches* fonksiyonu kullanılmaktadır.

Son olarak bulunan tarihler ve şirket ismi kullanıcı girdisinden silinir ve soru olarak bir sonraki adıma gönderilir.

2) *Bilginin Getirimi*: Kullanıcının sorduğu sorunun cevabının seçilen bildirimlerde yer alan belgelerde bulunması ön plandadır. Seçili bildirim içerisinde yer alan *json* formatındaki bildirim içeriği ve *pdf* formatlarındaki bildirim ve ek belgeler üzerinde çalışmaktadır. Tüm sistem *json* formatıyla çalıştığı için *pdf* formatında bulunanlar belgeler de *json* formatında ele alınmaktadır.

Burada kullanılan ana metod TF-IDF olmuştur. Şekil 3 üzerinde de görüldüğü gibi TF-IDF içerisinde term frequency ve inverse document frequency denilen iki kısım yer almaktadır. Bir terimin veya kelimenin TF değeri, bir belgedeki toplam kelime sayısına kıyasla terimin bir belgede kaç kez görüldüğüdür. Bir terimin IDF değeri, tüm metin verisinin yer aldığı corpus denilen yerde terimi içeren belgelerin oranını yansıtır.

Term Frequency (TF):

$$TF(t, \text{text}) = \frac{\text{Number of times term } t \text{ appears in text}}{\text{Total number of terms in the text}}$$

Inverse Document Frequency (IDF):

$$IDF(t, \text{corpus}) = \log \left(\frac{\text{Total number of texts in corpus (corpus)}}{\text{Number of texts containing term } t} \right)$$

TF-IDF:

$$TF-IDF(t, \text{text}, \text{corpus}) = TF(t, \text{text}) \times IDF(t, \text{corpus})$$

Şekil 1: TF-IDF Formülleri. [6]

TF-IDF tüm metin verisini vektör haline getirir. Aynı şekilde kullanıcının sorusu da vektör haline getirilir. Ardından, bu vektörler üzerinden Şekil 2 de görülen kosinüs benzerliği uygulanır. Buradan gelen benzerlik skoruna göre tüm belgeler arasında en ilgili sayfalar ortaya çıkmış olur.

$$\text{Cosine Similarity}(A, B) = \frac{A \cdot B}{\|A\| \times \|B\|}$$

Şekil 2: Kosinüs Benzerliği. [6]

3) *Kullanıcı Arayüzü*: Bu çalışmada, kullanıcıların bildirimler ve sorular arasında hızlı ve verimli bir şekilde geçiş yapabilmeleri için PyQt6 kullanılarak geliştirilmiş bir kullanıcı arayüzü (UI) tasarlanmıştır. Arayüz, kullanıcıların sorularını girmesini ve sonuçları görmesini kolaylaştıran çeşitli bileşenlerden oluşmaktadır.

Ana Pencere: Ana pencere, minimum 800x600 piksel boyutlarında ve kullanıcı dostu bir yapı olarak bulunmaktadır. Ana pencere, merkezi bir widget olan *QWidget* içerir ve bu widget üzerine diğer tüm bileşenler yerleştirilmiştir.

Alakalı Sayfalar Alanı: Kullanıcının girdiği soru ile ilgili ekrana getirilen sayfanın tam olarak aranan sonucu içermeme

ihtimaline karşın, ya da kullanıcının sorusuyla alakalı birden fazla sayfa olması ihtimali nedeniyle kullanıcının sorduğu soruyla alakalı en fazla 10 sayfa olmak üzere diğer sayfaları combo box bileşeni vasıtasıyla seçebilmesi sağlanmaktadır.

Geçmiş Liste Kutusu: Kullanıcının daha önce sormuş olduğu soruların bir listesini göstermektedir ve önceden sorduğu soruları hızlıca sorgulamasına olanak sağlamaktadır.

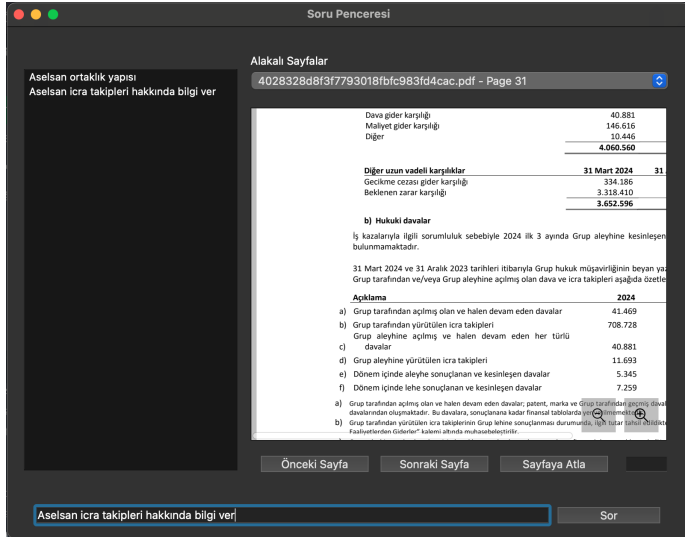
Sayfa Navigasyonu: Kullanıcıların belgelerde sayfalar arasında gezinmesine olanak tanıyan bu kısım, anlık olarak gösterilmekte olan pdf içerisinde önceki ve sonraki sayfalara gidilmesini sağlayan butonlar içermekte, aynı zamanda sayfa numarası girilebilmesi için bir metin kutusu ve girilen numadaki sayfanın direkt olarak getirilmesini sağlayan bir buton içermektedir.

Yükleniyor Etiketi: Bildirimlerin yüklenme sürecinde kullanıcıya bilgi vermektedir.

Soru Çerçevesi: Kullanıcının sorusunu girebileceği bir giriş kutusu ve soruyu gönderme butonu içeren bir çerçeve bulunmaktadır.

PDF Görüntüleyici: QPdfView bileşeni kullanılarak PDF belgelerinin görüntülenmesi sağlanmıştır. Kullanıcı bu alan içinde sayfanın tamamını ya da bir kısmını görünteleyebilmektedir. Yaklaştırma-uzaklaştırma işlevleri için iki buton bulunmakta olup, aynı zamanda kullanıcının gösterilen sayfayı kaydırmasına olanak sağlamaktadır

Bu bileşenler, kullanıcıların belgelerle etkileşime girmesini kolaylaştıracak şekilde düzenlenmiş ve her bileşenin pencere boyutuna göre dinamik olarak yeniden boyutlanması sağlanmıştır.



Şekil 3: Kullanıcı arayüzü. [6]

UI açıklaması: Yukarıdaki şekil, kullanıcı arayüzünün genel görünümünü göstermektedir. Ana bileşenler ve düzenleme mantığı bu şekil üzerinde belirtilmiştir. Kullanıcı, sol tarafta geçmiş soruları ve sonuçları görebilir, sağ üst köşede ilgili sayfalar arasında gezinme imkanı sahiptir, orta bölümde

PDF görüntüleyici bulunur ve alt kısımda yeni bir soru girişi yapılabilir.

IV. TESTLER

İlgili girdi/prompt UI kısmında girildiğinde aranan bilginin kaçınıcı alakalı sayfada yer aldığına dair birtakım test sonuçları Tablo I içinde yer almaktadır.

TABLO I: Testler

Girilen Prompt	Alakalı sayfalardan kaçınıcı istenilen sonucu içeriyor
Adel ocak 2024 faaliyet raporu hakkında bilgi istiyorum.	1
Adel mayıs 2024 tarihli rapordan yatırım faaliyetlerinden kaynaklanan nakit akışları hakkında bilgi istiyorum.	2
Adel ortaklık yapısı nasıldır?	5
Aselsan ortaklık yapısı hakkında bilgi istiyorum.	6
Aselsan ortaklık yapısı nasıldır?	3
Aselsan'ın kayıtlı sermaye tavanı ne kadardır?	1

V. SONUÇ

Bu çalışmada, Kamuyu Aydınlatma Platformu (KAP) üzerinden elde edilen finansal veriler kullanılarak kullanıcı sorularına yönelik bilgi getirimi yapılmıştır. Çalışmanın temel amacı, kullanıcıların finansal raporlar ve bildirimler gibi belgeler üzerinde sorduğu sorulara en ilgili sayfaları getirerek cevap verebilmektir.

Gerçekleştirilen testlerde, sistemin kullanıcı girdilerini doğru bir şekilde işlediği ve ilgili belgeleri yüksek doğruluk oranlarıyla getirdiği gözlemlenmiştir. Testler bölümünde sunulan tabloya göre, sistemin belirli sorulara karşılık gelen ilgili sayfaları başarılı bir şekilde tespit ettiği görülmüştür. Örneğin, "Adel ocak 2024 faaliyet raporu hakkında bilgi istiyorum." promptuna karşılık gelen sayfa sistem tarafından doğru olarak tespit edilmiştir.

Kullanıcıların çeşitli tarih, şirket ismi ve bildirim türü içeren sorularına karşılık sistemin verimli bir şekilde yanıt verdiği ve en alakalı sayfaları getirebildiği görülmüştür. Bu, geliştirilen bilgi getirimi sisteminin pratik kullanım açısından umut verici olduğunu göstermektedir.

Bazı durumlarda benzerlikten kaynaklı birtakım hatalı sonuçlar görülebilmektedir. Benzerlik kullanımında görülebilen bir durumdur. Bu hataların önlenmesi için hali hazırda bazı metotlar uygulanmıştır. Yine de bu metotlar %100 doğruluk verememektedir.

Sonuç olarak, geliştirilen sistemin finansal veri analizi ve bilgi getirimi alanında kullanıcıların ihtiyaçlarını karşılayabilecek düzeyde olduğu yapılan testlerle doğrulanmıştır.

KAYNAKLAR

- [1] H. S. ElSayed et al., "Chatbot as a Virtual Assistant to Retrieve Information from Datasheets Using Memory Controllers Domain Knowledge," 2023 30th IEEE International Conference on Electronics, Circuits and Systems (ICECS), Istanbul, Türkiye, 2023, pp. 1-7, doi: 10.1109/ICECS58634.2023.10382771.

- [2] Q. Zhi Lim, C. Poo Lee, K. Ming Lim and A. Kamsani Samangan, "UniRaG: Unification, Retrieval, and Generation for Multimodal Question Answering With Pre-Trained Language Models," in IEEE Access, vol. 12, pp. 71505-71519, 2024, doi: 10.1109/ACCESS.2024.3403101.
- [3] <https://www.kap.org.tr/tr/menu-icerik/KAP-Hakkinda/Genel-Bilgi>
- [4] <https://www.kap.org.tr/tr/bildirim-sorgu>
- [5] <https://github.com/parsee-ai/parsee-pdf-reader>
- [6] <https://medium.com/@anurag-jain/tf-idf-vectorization-with-cosine-similarity-eca3386d4423>